

ODE Solvers Using Bandlimited Approximations

G. Beylkin and K. Sandberg

*Department of Applied Mathematics
University of Colorado at Boulder
526 UCB
Boulder, CO 80309-0526*

Abstract

We use generalized Gaussian quadratures for exponentials to develop a new ODE solver. Nodes and weights of these quadratures are computed for a given bandlimit c and user selected accuracy ϵ , so that they integrate functions e^{ibx} , for all $|b| \leq c$, with accuracy ϵ . Nodes of these quadratures do not concentrate excessively near the end points of an interval as those of the standard, polynomial-based Gaussian quadratures. Due to this property, the usual implicit Runge Kutta (IRK) collocation method may be used with a large number of nodes, as long as the method chosen for solving the nonlinear system of equations converges. We show that the resulting ODE solver is symplectic and demonstrate (numerically) that it is A-stable. We use this solver, dubbed Band-limited Collocation (BLC-IRK), in the problem of orbit determination. Since BLC-IRK minimizes the number of nodes needed to obtain the solution, in this problem we achieve speed close to that of explicit multistep methods.

1 Introduction

Current methods for solving ODEs, be that multistep or Runge-Kutta, are based on polynomial approximations of functions. On the other hand, both recent and classical results [2,1,23,21,12,22] indicate that in many situations band-limited functions provide a near optimal tool for numerical integration and interpolation of functions. Using band-limited approximations, we construct a new method for solving the initial value problem for the ordinary

¹ This research was partially supported by AFOSR grant FA9550-07-1-0135, NSF grant DMS-0612358, DOE/ORNL grants 4000038129 and DE-FG02-03ER25583.

differential equations (ODEs) and demonstrate certain advantages of such approach. As an example, we consider orbit determination problem yielding a practical application as well as a gauge to ascertain the performance of new algorithms.

It is well-known that choosing between equally spaced and unequally spaced nodes on a specified time interval (i.e., choosing a multistep vs a collocation based Runge-Kutta method), results in significantly different properties of ODE solvers. When applied to solving a constant coefficient, linear system of ODEs (i.e., computing a matrix or scalar exponential, the so-called test problem), multistep schemes are A-stable only if their order does not exceed 2, the so-called Dahlquist barrier. In contrast, an A-stable implicit Runge-Kutta (IRK) scheme may be of arbitrary order. A class of A-stable IRK schemes uses the Gauss-Legendre quadrature nodes on each time interval; the order of such methods is 2ν , where ν is the number of nodes (see, e.g., [9]). Having $\{\mathcal{R}e(z) \leq 0, z \in \mathbb{C}\}$ as the region of absolute stability, assures that growth and decay of numerical solutions exactly mimics that of the analytic solutions of the test problem. A proper use of an A-stable scheme implies that the choice of parameters involves only accuracy consideration.

Another property of interest, that of preservation of volume in the phase space, identifies symplectic integrators. Symplectic integrators preserve a particular conserved quantity of nonlinear Hamiltonian systems as well as an approximate Hamiltonian. In problems of orbit determination, a symplectic integrator would keep the correct orbit more or less indefinitely with the error accumulating only in a position along the accurate orbit, thus closely reproducing a particular behavior of analytic solutions of nonlinear Hamiltonian systems. We note that IRK schemes which use the Gauss-Legendre nodes are symplectic (see, e.g., [9]).

While IRK schemes with the Gauss-Legendre nodes provide an excellent discretization of a system of ODEs, using many such nodes on a specified time interval presents a difficulty. It is well known that the nodes of the Gauss-Legendre quadratures (as well as any other polynomial based Gaussian quadratures) accumulate rapidly towards the end points of an interval. A heuristic reason for such accumulation is that these quadratures have to account for a possible rapid growth of polynomials near the boundary.

In this paper we demonstrate that, within IRK collocation methods, quadratures based on polynomials may be replaced by quadratures for band-limited exponentials. The nodes of these quadratures do not accumulate significantly toward the end points of an interval (a heuristic reason for an improved arrangement of nodes is that the exponentials do not grow anywhere in the interval). The quadratures for exponentials have been successfully used in problems of wave propagation [2] (see also [16]). We note that in problems of

wave propagation solutions are naturally well approximated by band-limited functions. It is also typical that solutions of ODEs are well approximated by band-limited exponentials. However, for ODEs there may be exceptions since solutions of some ODEs may, in fact, be polynomials or other functions that do not have an efficient approximation via exponentials and where polynomial-based quadratures are more effective. Thus, our method addresses numerical integration of ODEs whose solutions are well approximated by band-limited exponentials.

Unlike the classical Gaussian quadratures for polynomials which integrate exactly a subspace of polynomials up to a fixed degree, the Gaussian type quadratures for exponentials use a finite set of nodes to integrate an infinite set of functions, namely, $\{e^{ibx}\}_{|b|\leq c}$ on the interval $|x| \leq 1$. As there is no way to accomplish this exactly, these quadratures are constructed so that all exponentials with $|b| \leq c$ are integrated with accuracy of at least ϵ , where ϵ is arbitrarily small but finite. Such quadratures were constructed in [1] and, via a different approach in [23]. As observed in [2], quadrature nodes of this type do not concentrate excessively toward the end of the interval; the density of nodes increases toward the end points of the interval only by a factor that depends on the desired accuracy ϵ but not on the overall number of nodes.

Using quadratures integrating with accuracy ϵ^2 band-limited exponentials with bandlimit $2c$, we naturally arrive at a method for interpolation with accuracy ϵ of functions with bandlimit c (see [23,1]). It turns out that the nodes and weights of quadratures with interpolation accuracy $\epsilon \approx 10^{-15}$ can be constructed using only the standard double precision machine arithmetic. However, computing the integration matrix for accuracy $\epsilon \approx 10^{-15}$ does require an extended precision. We describe an approach that uses only quadruple precision for this purpose. Importantly, we use only the standard double precision when applying such quadratures and integration matrices within BLC-IRK method.

While analytically the classical Gauss-Legendre quadratures for polynomials are exact, in practice their accuracy is limited by the machine precision. By choosing (interpolation) accuracy $\epsilon \approx 10^{-15}$, our integrator is effectively “exact” within the double precision of machine arithmetic. Remarkably, using a particular construction of the integration matrix, we show that BLC-IRK method is (exactly) symplectic and, with high accuracy, A-stable. This result was unexpected and indicates that properties of approximate quadratures for band-limited exponentials need to be explored further.

While IRK schemes require solving a system of nonlinear equations at each time step, it does not automatically imply that such schemes are always computationally more expensive than explicit schemes. In the environment where the cost of function evaluation is high, the balance between the necessary it-

eration with fewer nodes of an implicit scheme vs significantly greater number of nodes of an explicit scheme (but no iteration), may tilt towards an implicit scheme. In problems of orbit determination, we use an additional observation that most iterations can be performed with an inexpensive (low fidelity) gravity model, making implicit schemes with a large number of nodes per time interval practical. We select the problem of orbit determination as an example where our approach is competitive with numerical schemes that are currently in use (see [3]). We take advantage of the reduced number of function calls to the full gravity model in a way that appears difficult to replicate using alternative schemes.

In order to accelerate solving a system of nonlinear equations, we modify the scheme by explicitly exponentiating the linear part of the force term. For the problem of orbit determination, this modification effectively makes use of the fact that the system is of the second order to accelerate convergence of iterations. So far we did not study possible acceleration of iterations using spectral deferred correction approach as in [6,14,8,10].

We start by providing background information on quadratures for band-limited functions in Section 2. We then describe BLC-IRK method in Section 3 (with some details deferred to Appendix). In Section 4 we detail our algorithm and provide examples.

2 Preliminaries: quadratures for band-limited functions

2.1 *Band-limited functions as a replacement of polynomials*

The quadratures constructed in [23,1] break with the conventional approach of using polynomials as the fundamental tool in analysis and computation. The approach based on polynomial approximations has a long tradition and leads to such notions as the order of convergence of numerical schemes, polynomial based interpolation, and so on. Recently, an alternative to polynomial approximations has been developed; it turns out that constructing quadratures for band-limited functions, e.g., exponentials e^{ibx} , with $|b| \leq c$, where c is the bandlimit, in many cases leads to significant improvement in performance of algorithms for interpolation, estimation and solving partial differential equations [2,16,11].

2.2 Bases for band-limited functions

It is well-known that a function whose Fourier Transform has compact support can not have compact support itself (unless it is identically zero). On the other hand, in physics duration of all signals is finite and their frequency response for all practical purposes is also limited. Thus, it is important to identify classes of functions which are essentially time and frequency limited. Towards this end, it is natural to analyze an operator whose effect on a function is to truncate it both in the original and the Fourier domains. Indeed, this has been the topic of a series of seminal papers by Slepian, Landau and Pollak, [22,12,13,18,19,20,21], where they observed (*inter alia*) that the eigenfunctions of such operator (see (2) below) are the Prolate Spheroidal Wave Functions (PSWFs) of classical Mathematical Physics.

While periodic band-limited functions may be expanded into Fourier series, neither the Fourier series nor the Fourier integral may be used efficiently for non-periodic functions on *intervals*. This motivates us to consider a class of functions admitting a representation via functions of the form $\{e^{ibx}\}_{|b|\leq c}$, $x \in [-1, 1]$, with a fixed parameter c (bandlimit). Following [1], let us consider the linear space of functions

$$\mathcal{E}_c = \left\{ u \in L^\infty([-1, 1]) \mid u(x) = \sum_{k \in \mathbb{Z}} a_k e^{icb_k x} : \{a_k\}_{k \in \mathbb{Z}} \in l^1, b_k \in [-1, 1] \right\}.$$

Given a finite accuracy ϵ , we represent the functions in \mathcal{E}_c by a fixed set of exponentials $\{e^{ic\tau_k x}\}_{k=1}^M$, where M is as small as possible. It turns out that by finding quadrature nodes $\{\tau_k\}_{k=1}^M$ and weights $\{w_k\}_{k=1}^M$ for exponentials with bandlimit $2c$ and accuracy ϵ^2 , we in fact obtain (with accuracy ϵ) a basis for \mathcal{E}_c with bandlimit c [1].

The generalized Gaussian quadratures for exponentials are constructed in [1] (see [23] for a different construction), which we summarize as

Lemma 1 *For $c > 0$ and any $\epsilon > 0$, there exist nodes $-1 < \tau_1 < \tau_2 < \dots < \tau_M < 1$ and corresponding weights $w_k > 0$, such that for any $x \in [-1, 1]$,*

$$\left| \int_{-1}^1 e^{ictx} dt - \sum_{k=1}^M w_k e^{ic\tau_k x} \right| < \epsilon, \quad (1)$$

where the number of nodes, M , is (nearly) optimal. The nodes and weights maintain the natural symmetry, $\tau_k = -\tau_{M-k+1}$ and $w_k = w_{M-k+1}$.

Remark 2 *The construction in [1] is more general and yields quadratures for band-limited exponentials integrated with a weight function. If the weight function is 1 as in Lemma 1, then the approach in [1] identifies the nodes of*

the generalized Gaussian quadratures in (1) as zeros of the discrete prolate spheroidal sequences (DPSSs) [20], corresponding to small eigenvalues.

Let us now consider band-limited functions,

$$\mathcal{B}_c = \{f \in L^2(\mathbb{R}) \mid \hat{f}(\omega) = 0 \text{ for } |\omega| \geq c\},$$

and briefly summarize some of the results in [22,12,13,18,21]. Let us define the operator $F_c : L^2[-1, 1] \rightarrow L^2[-1, 1]$,

$$F_c(\psi)(\omega) = \int_{-1}^1 e^{icx\omega} \psi(x) dx, \quad (2)$$

where $c > 0$ is the bandlimit. We also consider the operator $Q_c = \frac{c}{2\pi} F_c^* F_c$,

$$Q_c(\psi)(y) = \frac{1}{\pi} \int_{-1}^1 \frac{\sin(c(y-x))}{y-x} \psi(x) dx. \quad (3)$$

The eigenfunctions $\psi_0^c, \psi_1^c, \psi_2^c, \dots$ of Q_c coincide with those of F_c , and the eigenvalues μ_j of Q_c are related to the eigenvalues λ_j of F_c as

$$\mu_j = \frac{c}{2\pi} |\lambda_j|^2, \quad j = 0, 1, 2, \dots \quad (4)$$

While all $\mu_j < 1$, $j = 0, 1, \dots$, for large c the first approximately $2c/\pi$ eigenvalues μ_j are close to 1. They are followed by $\mathcal{O}(\log c)$ eigenvalues which decay exponentially fast forming a transition region; the rest of the eigenvalues μ_j are very close to zero.

The key result in [22] states that there exists a strictly increasing sequence of real numbers $\gamma_0 < \gamma_1 < \dots$, such that ψ_j^c are eigenfunctions of the differential operator,

$$L_c \psi_j^c \equiv \left(-(1-x^2) \frac{d^2}{dx^2} + 2x \frac{d}{dx} + c^2 x^2 \right) \psi_j^c(x) = \gamma_j \psi_j^c(x). \quad (5)$$

The eigenfunctions of L_c have been known as the angular prolate spheroidal functions (PSWF) before the connection with (2) was discovered in [22] (by demonstrating that L_c and F_c commute). We note that if $c \rightarrow 0$, then it follows from (5) that, in this limit, ψ_j^c become the Legendre polynomials. In many respects, PSWFs are strikingly similar to orthogonal polynomials; they are orthonormal, constitute a Chebychev system, and admit a version of Gaussian quadratures [23].

Since the space \mathcal{E}_c is dense in \mathcal{B}_c (and vice versa) [1], we note that the quadratures in [23] may potentially be used for the purposes of this paper as well (the nodes of the quadratures in [23] and those used in this paper are close but are not identical). Importantly, given accuracy ϵ , the functions

$\psi_0^c, \psi_1^c, \psi_2^c, \dots, \psi_{M-1}^c$ may be used as a basis for interpolation on the interval $[-1, 1]$ with $\tau_1, \tau_2, \dots, \tau_M$ as the interpolation nodes, provided that these are quadrature nodes constructed for the bandlimit $2c$ and accuracy ϵ^2 . Given functions $\psi_0^c, \psi_1^c, \psi_2^c, \dots, \psi_{M-1}^c$, we can construct an analogue of Lagrange interpolating polynomials, $R_k^c(x) = \sum_{j=0}^{M-1} \alpha_{kj} \psi_j^c(x)$, $x \in [-1, 1]$, by solving

$$\delta_{kl} = R_k^c(\tau_l) = \sum_{j=0}^{M-1} \alpha_{kj} \psi_j^c(\tau_l) \quad (6)$$

for the coefficients α_{kj} . The matrix $\psi_j^c(\tau_l)$ in (6) is well conditioned.

A well-known problem associated with the numerical use of orthogonal polynomials is concentration of their roots near the ends of the interval. Let us consider the ratio

$$r(M, \epsilon) = \frac{\tau_2 - \tau_1}{\tau_{\lfloor M/2 \rfloor} - \tau_{\lfloor M/2 \rfloor - 1}}, \quad (7)$$

where “ $\lfloor M/2 \rfloor$ ” denotes the least integer part, and look at it as a function of M . Observing that the distance between nodes of Gaussian quadratures for exponentials changes monotonically from the middle of an interval toward its end points, and that the smallest distance occurs between the nodes closest to the end point, the ratio (7) may be used as a measure of node accumulation. For example, the distance between the nodes near the end points of the standard Gaussian quadratures for polynomials decreases as $\mathcal{O}(1/M^2)$, so that we have $r(M, \epsilon) = \mathcal{O}(1/M)$, where M is the number of nodes. In Figure 1 we illustrate the behavior of $r(M, \epsilon)$ for the nodes of quadratures for band-limited exponentials. This ratio approaches a constant that depends on the accuracy ϵ but does not depend on the number of nodes.

Another important property of quadratures for exponentials emerges if we compare the critical sampling rate of a smooth *periodic* function, to that of smooth *non-periodic* function defined on an interval. Considering bandlimit c as a function of the number of nodes, M , and the desired accuracy ϵ , we observe that the oversampling factor,

$$\alpha(M, \epsilon) = \frac{\pi M}{c(M, \epsilon)} > 1,$$

approaches 1 for large M . We recall that in the case of the Gaussian quadratures for polynomials, this oversampling factor approaches $\frac{\pi}{2}$ rather than 1 (see e.g. [7]).

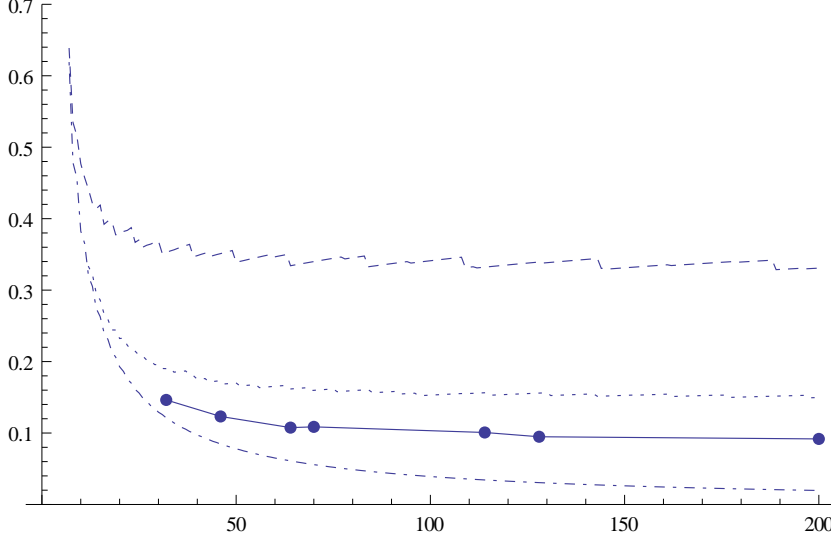


Figure 1. The ratio $r(M, \epsilon)$ in (7) as a function of the number of nodes M and interpolation accuracy $\epsilon \approx 10^{-3.5}$ (top curve, dashed), $\epsilon \approx 10^{-8.5}$ (middle curve, dotted) and $\epsilon \approx 10^{-13}$ (middle curve, solid). The dots on the solid curve indicate the number of nodes of quadratures used in our numerical experiments. The bottom curve shows this ratio for the Gauss-Legendre nodes.

2.3 Interpolating bases for band-limited functions

A basis of interpolating band-limited functions for the bandlimit c and accuracy ϵ plays the same role in the derivation of a system of nonlinear equations for solving ODEs as the bases of Lagrange interpolating polynomials defined on the Gauss-Legendre nodes. While (6) relies on available solutions of the differential equation (5), interpolating basis functions may also be obtained by solving the integral equation (2) (see [1]).

We start by first constructing a quadrature for the bandlimit $2c > 0$ and accuracy threshold $\epsilon^2 > 0$, yielding M nodes $\{\tau_m\}_{m=1}^M$ and weights $\{w_m\}_{m=1}^M$. For the inner product of two functions $f, g \in \mathcal{E}_c$, we have

$$\left| \int_{-1}^1 f(t)g(t)dt - \sum_{m=1}^M w_m f(\tau_m)g(\tau_m) \right| \leq \epsilon^2.$$

Following [1], we discretize (2) using nodes $\{\tau_m\}_{m=1}^M$ and weights $\{w_m\}_{m=1}^M$ and obtain an algebraic eigenvalue problem,

$$\sum_{l=1}^M w_l e^{ic\tau_m \tau_l} \Psi_j(\tau_l) = \eta_j \Psi_j(\tau_m). \quad (8)$$

The approximate PSWFs on $[-1, 1]$ are then defined consistent with (2) as

$$\Psi_j(x) = \frac{1}{\eta_j} \sum_{l=1}^M w_l e^{icx\tau_l} \Psi_j(\tau_l), \quad (9)$$

where η_j are the eigenvalues and $\Psi_j(\tau_l)$ the eigenvectors in (8). Following [1], we then define the interpolating basis for band-limited functions as

$$R_k(x) = \sum_{l=1}^M r_{kl} e^{ic\tau_l x}, \quad k = 1, \dots, M, \quad (10)$$

where

$$r_{kl} = \sum_{j=1}^M w_k \Psi_j(\tau_k) \frac{1}{\eta_j} \Psi_j(\tau_l) w_l. \quad (11)$$

It is shown in [1] that the functions $R_k(x)$ are interpolating, $R_k(\tau_l) = \delta_{kl}$.

3 BLC-IRK method

3.1 Discretization of Picard integral equation

We consider the initial value problem for a system of ODEs,

$$\mathbf{y}' = \mathbf{f}(t, \mathbf{y}), \quad \mathbf{y}(0) = \mathbf{y}_0,$$

or, equivalently,

$$\mathbf{y}(t) = \mathbf{y}_0 + \int_0^t \mathbf{f}(s, \mathbf{y}(s)) ds. \quad (12)$$

It is sufficient to discretize (12) on the interval $[0, t]$ since, by shifting the time variable, the initial condition may always be set at $t = 0$. We require

$$\mathbf{y}'(t\tau_j) = \mathbf{f}(t\tau_j, \mathbf{y}(t\tau_j)), \dots j = 1, \dots, M,$$

where $\{\tau_j\}_{j=1}^M$ are Gaussian nodes for band-limited exponentials on $[0, 1]$ (constructed for an appropriate bandlimit c and accuracy ϵ). We approximate

$$\|\mathbf{f}(t\tau, \mathbf{y}(t\tau)) - \sum_{j=1}^M \mathbf{f}(t\tau_j, \mathbf{y}(t\tau_j)) R_j(\tau)\| \leq \epsilon, \quad \tau \in [0, 1] \quad (13)$$

where $R_j(\tau)$ are interpolating basis functions associated with these quadratures and briefly described in Section 2.3 (see [1,2] for details). Using (13), we replace \mathbf{f} in (12) and evaluate $\mathbf{y}(t\tau)$ at the quadrature nodes yielding a

nonlinear system,

$$\mathbf{y}(t\tau_i) = \mathbf{y}_0 + \sum_{j=1}^M \mathbf{f}(t\tau_j, \mathbf{y}(t\tau_j)) \int_0^{t\tau_i} R_j(s) ds = \mathbf{y}_0 + \sum_{j=1}^M S_{ij} \mathbf{f}(t\tau_j, \mathbf{y}(t\tau_j)), \quad i = 1, \dots, M, \quad (14)$$

where $S_{ij} = \int_0^{t\tau_i} R_j(s) ds$ is the integration matrix. Solving for $\{\mathbf{y}(t\tau_j)\}_{j=1}^M$, we have from (12)

$$\mathbf{y}(t) = \mathbf{y}_0 + \sum_{j=1}^M w_j \mathbf{f}(t\tau_j, \mathbf{y}(t\tau_j)), \quad (15)$$

where $\{w_j\}_{j=1}^M$ are the quadrature weights. The result is an implicit Runge-Kutta method (IRK) where the usual Gauss-Legendre quadratures are replaced by Gaussian quadratures for band-limited exponentials.

The nodes, weights, and the entries of the integration matrix are typically organized in the Butcher tableau,

$$\begin{array}{c|c} \tau & S \\ \hline & w^t \end{array}.$$

Unlike in the standard IRK method based on Gauss-Legendre quadratures, we solve (14) on a time interval containing a large number of quadrature nodes, since these nodes do not concentrate excessively near the end points. This implies that the interval $[0, t]$ may be selected to be large in comparison with the usual choices in RK methods.

3.2 Exact Linear Part

In many problems (including that of orbit determination), the right hand side of the ODE, $\mathbf{f}(t, \mathbf{y})$, may be split into a linear and nonlinear parts,

$$\mathbf{f}(t, \mathbf{y}(t)) = \mathbf{L}\mathbf{y}(t) + \mathbf{g}(t, \mathbf{y}(t)),$$

so that the integral equation (12) may be written as

$$\mathbf{y}(t) = e^{t\mathbf{L}} \mathbf{y}_0 + \int_0^t e^{(t-s)\mathbf{L}} \mathbf{g}(s, \mathbf{y}(s)) ds. \quad (16)$$

If the operator $e^{t\mathbf{L}}$ can be computed efficiently, this formulation leads to savings when solving the integral equation iteratively.

We discretize (16) by using (13) and obtain

$$\begin{aligned}
\mathbf{y}(t\tau_i) &= e^{t\tau_i \mathbf{L}} \mathbf{y}_0 + \sum_{j=1}^M e^{t(\tau_i - \tau_j) \mathbf{L}} \mathbf{g}(t\tau_j, \mathbf{y}(t\tau_j)) \int_0^{\tau_i} R_j(s) ds \\
&= e^{t\tau_i \mathbf{L}} \mathbf{y}_0 + \sum_{j=1}^M S_{ij} e^{t(\tau_i - \tau_j) \mathbf{L}} \mathbf{g}(t\tau_j, \mathbf{y}(t\tau_j))
\end{aligned} \tag{17}$$

where $S_{ij} = \int_0^{\tau_i} R_j(s) ds$. We note that (14) is a special case of (17) with $\mathbf{L} = 0$ and $\mathbf{g} = \mathbf{f}$.

3.3 Symplectic integrators

Following [17], let us introduce matrix $\mathcal{M} = \{m_{kj}\}_{k,j=1}^M$ for an M -stage IRK scheme,

$$m_{kj} = w_k S_{kj} + w_j S_{jk} - w_k w_j, \tag{18}$$

where the weights $w = \{w_k\}_{k=1}^M$ and the integration matrix $S = \{S_{kj}\}_{k,j=1}^M$ define the Butcher's tableau for the method.

It is shown in [17] that

Theorem 3 *If matrix $\mathcal{M} = 0$ in (18), then an M -stage IRK scheme is symplectic.*

This condition, $\mathcal{M} = 0$, is satisfied for the Gauss-Legendre RK methods, see e.g. [4,17]. We enforce this condition for BLC-IRK method by an $\mathcal{O}(\epsilon^2)$ modification of the weights and of the integration matrix. For convenience, we consider the band-limited exponentials and integration matrix on the interval $[-1, 1]$ (rather than $[0, 1]$ usually used for ODEs).

Proposition 4 *Let $\{\tau_j\}_{j=1}^M$ and $\{w_j\}_{j=1}^M$ be quadrature nodes and weights for the bandlimit $2c$ and accuracy ϵ^2 . Consider interpolating basis functions on these quadrature nodes, $R_i(\tau)$, $R_i(\tau_j) = \delta_{ij}$, $i, j = 1, \dots, M$, and define $F_i(\tau) = \int_{-1}^{\tau} R_i(s) ds$. Then we have*

$$\left| \int_{-1}^1 F_j(\tau) F_i'(\tau) d\tau - \sum_{k=1}^M w_k F_j(\tau_k) F_i'(\tau_k) \right| < \epsilon^2 \tag{19}$$

or

$$\left| \int_{-1}^1 \left(\int_{-1}^{\tau} R_j(s) ds \right) R_i(\tau) d\tau - w_i \int_{-1}^{\tau_i} R_j(s) ds \right| < \epsilon^2,$$

and

$$\left| \int_{-1}^1 R_i(\tau) d\tau - \sum_{k=1}^M w_k R_i(\tau_k) \right| < \epsilon^2, \tag{20}$$

or

$$\left| \int_{-1}^1 R_i(s) ds - w_i \right| < \epsilon^2.$$

PROOF. The relations in (19) and (20) is the property of the quadrature, since the bandlimit of the product $F_j(\tau)F'_i(\tau)$ is less or equal to $2c$ and that of $R_i(\tau)$ is less or equal to c . Due to the interpolating property of $R_i(\tau)$, we have

$$\sum_{k=1}^M w_k F_j(\tau_k) F'_i(\tau_k) = \sum_{k=1}^M \left(\int_{-1}^{\tau_k} R_j(s) ds \right) w_k R_i(\tau_k) = w_i \int_{-1}^{\tau_i} R_j(s) ds \quad (21)$$

and

$$\sum_{k=1}^M w_k R_i(\tau_k) = w_i$$

Also, by definition,

$$\int_{-1}^1 F_j(\tau) F'_i(\tau) d\tau = \int_{-1}^1 \left(\int_0^\tau R_j(s) ds \right) R_i(\tau) d\tau,$$

and the result follows.

Theorem 5 Let $\{\tau_j\}_{j=1}^M$ be quadrature nodes of the quadrature for the bandlimit $2c$ and accuracy ϵ^2 and $R_i(\tau)$, $R_i(\tau_j) = \delta_{ij}$, $i, j = 1, \dots, M$, the corresponding interpolating basis. Let us define weights for the quadrature as

$$w_k = \int_{-1}^1 R_k(\tau) d\tau \quad (22)$$

and the integration matrix as

$$S_{kl} = \frac{\int_{-1}^1 \left(\int_{-1}^\tau R_l(s) ds \right) R_k(\tau) d\tau}{w_k}, \quad k, l = 1, \dots, M. \quad (23)$$

Then

$$w_k S_{kl} + w_l S_{lk} - w_k w_l = 0, \quad (24)$$

and the implicit scheme using these nodes and weights is symplectic.

PROOF. Using the propositions above, we observe that the weights defined in (22) are the same (up to accuracy ϵ^2) as those of the quadrature. The result follows by setting $F_k(\tau) = \int_{-1}^\tau R_k(\tau) d\tau$, $F'_k(\tau) = R_k(\tau)$ and integrating by parts to obtain

$$\begin{aligned} w_k S_{kl} + w_l S_{lk} - w_k w_l &= \int_{-1}^1 F_l(\tau) F'_k(\tau) d\tau + \int_{-1}^1 F_k(\tau) F'_l(\tau) d\tau - w_k w_l \\ &= F_l(1) F_k(1) - w_k w_l. \end{aligned}$$

By the definition of the weights, we have $F_k(1) = w_k$ and, hence, $F_l(1) F_k(1) - w_k w_l = 0$.

3.4 Construction of the integration matrix

There are at least three approaches to compute the integration matrix. Two of them, presented in Appendix, rely on Theorem 5 and differ in the construction of interpolating basis functions. In what appears to be a simpler approach, the integration matrix may be obtained without computing interpolating basis functions explicitly and, instead, using a collocation condition together with the symplectic condition (24).

Besides satisfying (24), we require that the method accurately solves the test problem

$$y' = ic\tau_k y, \quad y(-1) = 0$$

on the interval $[-1, 1]$, where τ_k are the nodes of the quadrature, $k = 1, \dots, M$. To find solution of this equation, $y_k(t) = e^{ic\tau_k t} - e^{-ic\tau_k}$, we require (see (14)) that at the nodes, $t = \tau_j$,

$$\left| \frac{e^{ic\tau_k \tau_j} - e^{-ic\tau_k}}{ic\tau_k} - \sum_{m=1}^M S_{jm} e^{ic\tau_k \tau_m} \right| \leq \epsilon, \quad (25)$$

and compute the integration matrix as the solution of (24) satisfying an approximate collocation condition (25).

Defining the symmetric part of the integration matrix

$$T_{jm} = \frac{w_j w_m}{w_j + w_m}, \quad (26)$$

we set

$$S_{jm} = T_{jm} + A_{jm} w_m, \quad (27)$$

so that it follows from (24) that A_{jm} is anti-symmetric,

$$A_{jm} + A_{mj} = 0.$$

From (25), we obtain

$$\frac{e^{ic\tau_k \tau_j} - e^{-ic\tau_k}}{ic\tau_k} - \sum_{m=1}^M T_{jm} e^{ic\tau_k \tau_m} = \sum_{m=1}^M A_{jm} w_m e^{ic\tau_k \tau_m}. \quad (28)$$

We split the real and imaginary parts

$$\frac{e^{ic\tau_k \tau_j} - e^{-ic\tau_k}}{ic\tau_k} - \sum_{m=1}^M T_{jm} e^{ic\tau_k \tau_m} = u_{jk} + iv_{jk},$$

where

$$u_{jk} = (\tau_j + 1) \operatorname{sinc}(c\tau_k(\tau_j + 1)/2) \cos(c\tau_k(\tau_j - 1)/2) - \sum_{m=1}^M T_{jm} \cos(c\tau_k\tau_m)$$

and, since $T_{jm} = T_{j(M-m+1)}$ due to symmetry of the weights,

$$v_{jk} = (\tau_j + 1) \operatorname{sinc}(c\tau_k(\tau_j + 1)/2) \sin(c\tau_k(\tau_j - 1)/2).$$

Splitting the real and imaginary parts in (28), we have

$$u_{jk} = \sum_{m=1}^M A_{jm} w_m \cos(c\tau_k\tau_m), \quad v_{jk} = \sum_{m=1}^M A_{jm} w_m \sin(c\tau_k\tau_m).$$

Since matrices $\cos(c\tau_k\tau_m)$ and $\sin(c\tau_k\tau_m)$ are rank deficient, we choose to combine these equations as follows

$$u_{jk} + v_{jk} = \sum_{m=1}^M A_{jm} w_m (\cos(c\tau_k\tau_m) + \sin(c\tau_k\tau_m)). \quad (29)$$

Solving this equation (using quadruple precision since this system is ill-conditioned), we find matrix \tilde{A} and discover that, while $S_{jm} = T_{jm} + \tilde{A}_{jm} w_m$ makes (25) into an equality, it is not exactly anti-symmetric. We then anti-symmetrize this matrix by averaging, $A_{jm} = -A_{mj} = (\tilde{A}_{jm} - \tilde{A}_{mj})/2$. Combining so obtained anti-symmetric matrix A with (26) via (27), we obtain the integration matrix. We then verify that the inequality (25) is satisfied.

Remark 6 *The fact that integration matrix satisfies (24) and the inequality (25) indicates that, perhaps by a slight modification of nodes and weights of the quadrature, it might be possible to satisfy (24) and (25) with $\epsilon = 0$.*

3.5 A-stability of the BLC-IRK method

As shown in e.g. [9, Section 4.3], in order to ascertain stability of an IRK method, it is sufficient to consider a rational function

$$r(z) = 1 + z\mathbf{w}^t(I - zS)^{-1}\mathbf{1}, \quad (30)$$

where S is the integration matrix, \mathbf{w} is a vector of weights and $\mathbf{1}$ is a vector with all entries set to 1, and verify that $|r(z)| \leq 1$ in the left half of the complex plane, $\operatorname{Re}(z) \leq 0$. This function is an approximation of the solution e^z of the test problem

$$y' = zy, \quad y(0) = 1$$

computed via (14) and (15) on the interval $[0, 1]$. If all poles of $r(z)$ have a positive real part, then it is sufficient to verify this inequality only on the

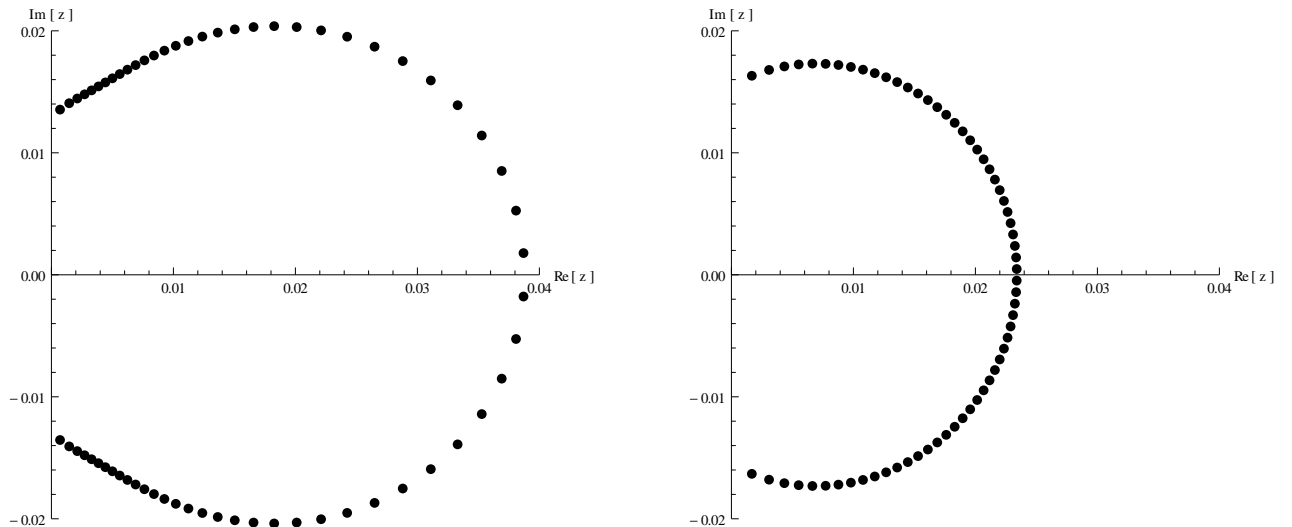


Figure 2. Eigenvalues (computed using quadruple precision) of the integration matrix for BLC-IRK scheme with 64 nodes corresponding to the bandlimit 17π and, for comparison, eigenvalues of the integration matrix of the standard IRK scheme 64 Gauss-Legendre nodes.

imaginary axis, $z = iy$, $y \in \mathbb{R}$. In fact, it may be possible to show that $r(z)$ is unimodular on imaginary axis, $|r(iy)| = 1$, for $y \in \mathbb{R}$. Implicit Runge-Kutta methods based on Gauss-Legendre nodes are A-stable (see e.g [9]) and, indeed, for these methods $r(z)$ is unimodular on imaginary axis.

Given an $M \times M$ matrix S with M_1 complex eigenvalues and M_2 real eigenvalues, if function (30) is unimodular on the imaginary axis then it has a particular form,

$$r(z) = \prod_{k=1}^{M_1} \frac{z + \bar{\lambda}_k^{-1}}{z - \lambda_k^{-1}} \frac{z + \lambda_k^{-1}}{z - \bar{\lambda}_k^{-1}} \prod_{k'=1}^{M_2} \frac{z + \lambda_{k'}^{-1}}{z - \lambda_{k'}^{-1}}, \quad (31)$$

where $2M_1 + M_2 = M$. Currently, we do not have an analytic proof of A-stability of BLC-IRK method; instead we verify this property numerically. We compute eigenvalues of the integration matrix to obtain the poles of $r(z)$ and check that all eigenvalues have a positive real part separated from zero. For example, integration matrix for BLC-IRK method with 64 nodes (bandlimit $c = 17\pi$) has all eigenvalues with real part larger than $0.7 \cdot 10^{-3}$. One way to check that $r(z)$ has the form (31) is to compute $r(-\bar{\lambda}_k^{-1})$ and $r(-\lambda_k^{-1})$ to observe if these are its zeros. In fact, it is the case with high (quadruple) precision.

One can argue heuristically that since a rational function with M poles has at most $2M$ real parameters (since matrix S is real its eigenvalues appear in complex conjugate pairs) and since, by construction, $r(iy)$ for $|y| \leq c$ is an accurate approximation to e^{iy} (which is obviously unimodular), $r(z)$ is then unimodular. It remains to show it rigorously; a possible proof may depend on demonstrating a conjecture in Remark 6.

4 Applications

4.1 Algorithm

A fixed point iteration to solve (17) is the usual choice (as long as it converges). Let N_{it} denote the number of iterations, which can either be set to a fixed number or determined adaptively. Labeling the intermediate solutions in the iteration scheme as $\mathbf{y}^{(i)}$, $i = 1, \dots, N_{it}$, we have

- (1) Initialize $y^{(1)}(t\tau_i) = \mathbf{y}_0$, $i = 1, \dots, M$.
- (2) **For** $k = 1, \dots, N_{it}$
For $m = 1, \dots, M$

- (a) Update the solution at the node m :

$$\mathbf{y}^{(k)}(t\tau_m) = e^{t\tau_m \mathbf{L}} \mathbf{y}_0 + \sum_{j=1}^M S_{mj} e^{t(\tau_m - \tau_j) \mathbf{L}} \mathbf{g}(t\tau_j, \mathbf{y}^{(k)}(t\tau_j))$$

- (b) Update the right hand side: $\mathbf{g}(t\tau_m, \mathbf{y}^{(k)}(t\tau_m))$

We note that the updated value of $y^{(k)}(t\tau_m)$ is used in the computation at the next node τ_{m+1} within the same iteration k . This is essential for a fast convergence.

Remark 7 *Although we currently apply the integration matrix directly, using a large time interval and, consequently, a large number of nodes per interval, opens a possibility of using fast algorithms for this purpose. Such algorithms may be faster than the direct application of the matrix only for a sufficiently large matrix size and are less efficient than the direct method if the size is relatively small. Since we may choose many nodes, it makes sense to ask if the integration matrix of an IRK type method may be applied in $\mathcal{O}(M)$ or $\mathcal{O}(M \log M)$ operations rather $\mathcal{O}(M^2)$. We mention an example of an algorithm for this purpose using the partitioned low rank (PLR) representation (as it was described in e.g., [2]) but leave open a possibility of more efficient approaches.*

4.2 Problem of Orbit Determination

Let us consider the spherical harmonic model of a gravitational potential of degree N ,

$$V^{(N)}(r, \theta, \lambda) = \frac{\mu}{r} \left(1 + \sum_{n=2}^N \left(\frac{R}{r} \right)^{-n} Y_n(\theta, \lambda) \right), \quad (32)$$

with

$$Y_n(\theta, \lambda) = \sum_{m=0}^n \bar{P}_n^m(\sin \theta) (\bar{C}_{nm} \cos(m\lambda) + \bar{S}_{nm} \sin(m\lambda)), \quad (33)$$

where \bar{P}_n^m are normalized associated Legendre functions and \bar{C}_{nm} and \bar{S}_{nm} are normalized gravitational coefficients. In case of the Earth's gravitational model, μ is the Earth's gravitational constant and R is chosen to be the Earth's equatorial radius. Choosing the Cartesian coordinates, we write $V^{(N)}(\mathbf{r})$, $\mathbf{r} = (x, y, z)$, assuming that the values $V^{(N)}(\mathbf{r})$ are evaluated via (32) by changing from the Cartesian to the spherical coordinates, $r = \sqrt{x^2 + y^2 + z^2}$, $\theta = \arcsin(z/r)$ and $\lambda = \arctan(y/x)$.

We formulate the system of ODEs in the Cartesian coordinates and denote the solution as $\mathbf{r}(t) = (x(t), y(t), z(t))$. Setting $\mathbf{G}^{(N)}(\mathbf{r}) = \nabla V^{(N)}(\mathbf{r})$, we consider the initial value problem

$$\frac{d^2}{dt^2} \mathbf{r}(t) = -\mathbf{G}^{(N)}(\mathbf{r}(t)), \quad \mathbf{r}(0) = \mathbf{r}_0 = \begin{pmatrix} x_0 \\ y_0 \\ z_0 \end{pmatrix}. \quad (34)$$

We observe that in gravitational models the first few terms are typically large compared to the rest of the terms. For example, in EGM96 gravitational model [15] the only non-zero coefficients for $Y_2(\theta, \lambda)$ are \bar{C}_{20} , \bar{C}_{22} and \bar{S}_{22} , where $\bar{C}_{20} \approx -0.48 \cdot 10^{-3}$, $\bar{C}_{22} \approx 0.24 \cdot 10^{-5}$, and $\bar{S}_{22} \approx -0.14 \cdot 10^{-5}$, whereas all other coefficients are less than $0.14 \cdot 10^{-5}$. For this reason it makes sense to split the force as

$$\mathbf{G}^{(N)}(\mathbf{r}) = \mathbf{G}^{(2)}(\mathbf{r}) + (\mathbf{G}^{(N)}(\mathbf{r}) - \mathbf{G}^{(2)}(\mathbf{r}))$$

and use only $\mathbf{G}^{(2)}(\mathbf{r})$ in most of the iterations (since using the full model, $\mathbf{G}^{(N)}(\mathbf{r})$, may be expensive).

We first use the gravity model of degree $N = 2$ on a large portion of an orbit (e.g., 1/2 of a period) to solve the system of nonlinear equations via fixed point iteration. Once the approximate solution $\tilde{\mathbf{r}}(t)$ to

$$\frac{d^2}{dt^2} \tilde{\mathbf{r}}(t) = -\mathbf{G}^{(2)}(\tilde{\mathbf{r}}(t)), \quad \tilde{\mathbf{r}}(0) = \mathbf{r}_0 = \begin{pmatrix} x_0 \\ y_0 \\ z_0 \end{pmatrix},$$

is obtained, we then access the full gravity model $\mathbf{G}^{(N)}(\tilde{\mathbf{r}}(t\tau_j))$ to evaluate the forces at the nodes τ_j which, by now, are located close to their correct positions. We continue iteration (without accessing the full gravity model again) to adjust the orbit. This results in an essentially correct trajectory. At this

point we may (and currently do) access the full gravity model $\mathbf{G}^{(N)}$ one more time to evaluate the gravitational force and perform another iteration. Thus, we access the full gravity model at most twice per node while using a number of nodes that is substantially lower than in traditional methods.

Next, let us write the orbit determination problem in a form that conforms with the algorithm in Section 4.1. Effectively, we make use of the fact that system (34) is of the second order. We define the six component vector

$$\mathbf{u}(t) = \begin{bmatrix} \mathbf{r}(t) \\ \mathbf{r}'(t) \end{bmatrix} = \begin{bmatrix} \mathbf{r}(t) \\ \mathbf{v}(t) \end{bmatrix},$$

where $\mathbf{r}'(t) = \mathbf{v}(t)$ is the velocity, and the matrix

$$\mathbf{L} = \begin{pmatrix} \mathbf{0} & \mathbf{I} \\ \mathbf{0} & \mathbf{0} \end{pmatrix},$$

where \mathbf{I} is 3×3 identity matrix. We have

$$\frac{d}{dt} \begin{bmatrix} \mathbf{r}(t) \\ \mathbf{v}(t) \end{bmatrix} = \mathbf{L} \begin{bmatrix} \mathbf{r}(t) \\ \mathbf{v}(t) \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ -\mathbf{G}^{(N)}(\mathbf{r}(t)) \end{bmatrix}, \quad (35)$$

and the orbit determination problem is now given by (16) with appropriate forces as follow from (35). Using (16) accelerates convergence of the fixed point iteration in our scheme.

4.3 Example

We present an example of using our method. An extensive study of the method for applications in astrodynamics may be found in [3] and here we simply demonstrate that our scheme allows large time steps and requires relatively few evaluations of the full gravity model. As the cost of evaluating the full (high-degree) gravity model is substantial, this results in significant computational savings.

As an example, we simulate an orbit with initial condition

$$\mathbf{r}|_{t=0} = \begin{pmatrix} x_0 \\ y_0 \\ z_0 \end{pmatrix} = \begin{pmatrix} 2284.060 \\ 6275.400 \\ 4.431 \end{pmatrix} \text{ (km)}$$

and

$$\frac{d\mathbf{r}}{dt}|_{t=0} = \begin{pmatrix} -5.947 \\ 2.164 \\ 0 \end{pmatrix} \text{ (km/s),}$$

and propagate it for 86,000 seconds (approximately 1 day). We use 22 time intervals and, on each interval, quadratures with 74 nodes. Hence, on average, this corresponds to time distance between nodes of approximately 53 seconds. For the full gravitational model we use a 70 degree spherical harmonics model WGS84 [5].

We compute the orbit trajectory using the algorithm from Section 4.1 adopted to the problem of orbit propagation as described in Section 4.2. We compare the result with a trajectory generated by an 8th-order Gauss-Jackson integration scheme with one second time step.

Achieving an error of less than 5 cm at the final time, we need 6512 evaluations of the reduced (3-term) gravitational model, and 3256 evaluations of the full gravitational model.

5 Conclusions

We have constructed an implicit, symplectic integrator that has speed comparable to explicit multistep integrators currently used for orbit computation. The key difference with the traditional IRK method is that our scheme uses quadratures for band-limited exponentials rather than the traditional Gaussian quadratures constructed for the orthogonal Legendre polynomials. The nodes of quadratures for band-limited exponentials do not concentrate excessively towards the end points of an interval thus removing a practical limit on the number of nodes used within each time interval.

6 Appendix

In both approaches described below we use nodes of generalized Gaussian quadratures for exponentials $\{\tau_l\}_{l=1}^M$ constructed in [1] (see Lemma 1). Some of the steps may require extended precision to yield accurate results.

6.1 Computing integration matrix using exact PSWFs

In this approach we assume that the solutions $\psi_j^c(x)$ and the eigenvalues λ_j satisfying

$$(F_c \psi_j^c)(x) = \int_{-1}^1 e^{icxy} \psi_j^c(y) dy = \lambda_j \psi_j^c(x), \quad (36)$$

where F_c is defined in (2), are available. We use (6) and the matrix of values of PSWFs at the nodes, $\psi_j^c(\tau_l)$, to compute coefficients α_{kj} , so that we have

$$R_k^c(\tau) = \sum_{j=0}^{M-1} \alpha_{kj} \psi_j^c(\tau), \quad k = 1, \dots, M.$$

We then compute weights using (22),

$$w_k = \int_{-1}^1 R_k^c(x) dx = \sum_{j=0}^{M-1} \alpha_{kj} \int_{-1}^1 \psi_j^c(x) dx = \sum_{j=0}^{M-1} \alpha_{kj} \lambda_j \psi_j^c(0).$$

Next we define

$$F_l^c(x) = \int_{-1}^x R_l^c(s) ds = \sum_{j=0}^{M-1} \alpha_{lj} \int_{-1}^x \psi_j^c(s) ds = \sum_{j=0}^{M-1} \alpha_{lj} \Phi_j^c(x),$$

where

$$\Phi_j^c(x) = \int_{-1}^x \psi_j^c(s) ds. \quad (37)$$

In order to compute the integration matrix (23), we need to evaluate

$$w_k S_{kl} = \int_{-1}^1 F_l^c(x) R_k^c(x) dx = \sum_{j,j'=0}^{M-1} \alpha_{lj} \alpha_{kj'} \int_{-1}^1 \Phi_j^c(x) \psi_{j'}^c(x) dx = \sum_{j,j'=0}^{M-1} \alpha_{lj} \alpha_{kj'} I_{jj'},$$

where

$$I_{jj'} = \int_{-1}^1 \Phi_j^c(x) \psi_{j'}^c(x) dx = \int_{-1}^1 \Phi_j^c(x) \frac{d}{dx} \Phi_{j'}^c(x) dx. \quad (38)$$

We have

Proposition 8 *If j and j' are both even, then*

$$I_{jj'} = I_{j'j} = \frac{1}{2} \lambda_j \lambda_{j'} \psi_j^c(0) \psi_{j'}^c(0). \quad (39)$$

If j and j' are both odd, then

$$I_{jj'} = 0. \quad (40)$$

If j is even and j' is odd, then

$$I_{jj'} = -I_{j'j}, \quad (41)$$

$$I_{jj'} = \frac{\lambda_{j'}}{ic\lambda_j} \int_{-1}^1 \psi_j^c(y) \frac{\psi_{j'}^c(y)}{y} dy \quad (42)$$

and

$$I_{j'j} = \frac{\lambda_j}{ic\lambda_{j'}} \left(\int_{-1}^1 \psi_j^c(y) \frac{\psi_{j'}^c(y)}{y} dy - 2\psi_j^c(0) \int_0^1 \frac{\psi_{j'}^c(y)}{y} dy + ic\psi_j^c(0) \bar{\lambda}_{j'} \int_0^1 \psi_{j'}^c(y) dy \right). \quad (43)$$

We use (42) if $|\lambda_{j'}| < |\lambda_j|$, (43) otherwise.

PROOF. Integrating (38) by parts, we obtain

$$I_{jj'} + I_{j'j} = \Phi_j^c(1)\Phi_{j'}^c(1) - \Phi_j^c(-1)\Phi_{j'}^c(-1) = \lambda_j\lambda_{j'}\psi_j^c(0)\psi_{j'}^c(0) \quad (44)$$

and, since $\psi_j(0) = 0$ if j is odd (due to parity of PSWFs), arrive at (40) and (41).

Using (37) and (36), we have

$$\Phi_j^c(x) = \frac{1}{\lambda_j} \int_{-1}^1 \left(\int_{-1}^x e^{icys} ds \right) \psi_j^c(y) dy = \frac{1}{\lambda_j} \int_{-1}^1 \frac{e^{icyx} - e^{-icy}}{icy} \psi_j^c(y) dy,$$

and, thus,

$$\begin{aligned} I_{jj'} &= \frac{1}{\lambda_j} \int_{-1}^1 \left[\int_{-1}^1 \frac{e^{icyx} - e^{-icy}}{icy} \psi_j^c(y) dy \right] \psi_{j'}^c(x) dx \\ &= \frac{\lambda_{j'}}{ic\lambda_j} \left(\int_{-1}^1 \psi_j^c(y) \frac{\psi_{j'}^c(y)}{y} dy - \psi_{j'}^c(0) \int_{-1}^1 \frac{\psi_j^c(y)}{y} e^{-icy} dy \right). \end{aligned} \quad (45)$$

It follows from (45) that if j is even and j' is odd (so that $\psi_{j'}^c(0) = 0$), we obtain (42) and

$$I_{j'j} = \frac{\lambda_j}{ic\lambda_{j'}} \left(\int_{-1}^1 \psi_j^c(y) \frac{\psi_{j'}^c(y)}{y} dy - \psi_j^c(0) \int_{-1}^1 \frac{\psi_{j'}^c(y)}{y} e^{-icy} dy \right).$$

Introducing

$$u(x) = \int_{-1}^1 \frac{\psi_{j'}^c(y)}{y} e^{-icyx} dy,$$

we have

$$u'(x) = -ic \int_{-1}^1 \psi_{j'}^c(y) e^{-icyx} dy = -ic\bar{\lambda}_{j'}\psi_{j'}^c(x)$$

so that

$$u(x) = u(a) - ic\bar{\lambda}_{j'} \int_a^x \psi_{j'}^c(s) ds.$$

Setting $x = 1$ and $a = 0$, we obtain

$$\begin{aligned}
\int_{-1}^1 \frac{\psi_{j'}^c(y)}{y} e^{-icy} dy &= \int_{-1}^1 \frac{\psi_{j'}^c(y)}{y} dy - ic\bar{\lambda}_{j'} \int_0^1 \psi_{j'}^c(y) dy \\
&= 2 \int_0^1 \frac{\psi_{j'}^c(y)}{y} dy - ic\bar{\lambda}_{j'} \int_0^1 \psi_{j'}^c(y) dy
\end{aligned}$$

and arrive at (43).

6.2 Computing integration matrix using approximate PSWFs

If the interpolating basis for band-limited functions is defined via (10), then the coefficients r_{kl} are obtained using

$$\delta_{km} = R_k(\tau_m) = \sum_{l=1}^M r_{kl} e^{ic\tau_l \tau_m} \quad (46)$$

by inverting the matrix $E = \{e^{ic\tau_l \tau_m}\}_{l,m=1,\dots,M}$. We have

$$F_k(x) = \int_{-1}^x R_k(s) ds = \sum_{l=1}^M r_{kl} \frac{e^{ic\tau_l x} - e^{-ic\tau_l}}{ic\tau_l}$$

and compute

$$\begin{aligned}
w_k S_{kl} &= \int_{-1}^1 F_l(x) R_k(x) dx \\
&= \sum_{j,j'=1,\dots,M} r_{kj} r_{lj'} \int_{-1}^1 e^{ic\tau_j x} \frac{e^{ic\tau_{j'} x} - e^{-ic\tau_{j'}}}{ic\tau_{j'}} dx \\
&= \sum_{j,j'=1,\dots,M} r_{kj} r_{lj'} G_{jj'},
\end{aligned}$$

where

$$G_{jj'} = 2 \frac{\text{sinc}(c(\tau_j + \tau_{j'})) - e^{-ic\tau_{j'}} \text{sinc}(c\tau_j)}{ic\tau_{j'}}.$$

Thus, we have

$$w_k S_{kl} = \left(E^{-1} G E^{-1} \right)_{kl}.$$

References

- [1] G. Beylkin and L. Monzón. On generalized Gaussian quadratures for exponentials and their applications. *Appl. Comput. Harmon. Anal.*, 12(3):332–373, 2002.

- [2] G. Beylkin and K. Sandberg. Wave propagation using bases for bandlimited functions. *Wave Motion*, 41(3):263–291, 2005.
- [3] B.K. Bradley, B.A. Jones, G. Beylkin, and P. Axelrad. A new numerical integration technique in astrodynamics. In *22nd Annual AAS/AIAA Space Flight Mechanics Meeting, Charleston, SC, Jan. 29 - Feb. 2, 2012*.
- [4] K. Dekker and J.G. Verwer. *Stability of the Runge-Kutta methods for stiff nonlinear differential equations*. North-Holland, Amsterdam, 1984.
- [5] Dept. of Defense World Geodetic System. Defense Mapping Agency Technical Report. Technical report, 1987. DMA TR 8350.2.
- [6] A. Dutt, L. Greengard, and V. Rokhlin. Spectral deferred correction methods for ordinary differential equations. *BIT*, 40(2):241–266, 2000.
- [7] D. Gottlieb and S. A. Orszag. *Numerical analysis of spectral methods: theory and applications*. Society for Industrial and Applied Mathematics, Philadelphia, Pa., 1977. CBMS-NSF Regional Conference Series in Applied Mathematics, No. 26.
- [8] J. Huang, J. Jia, and M. Minion. Accelerating the convergence of spectral deferred correction methods. *J. Comput. Phys.*, 214(2):633–656, 2006.
- [9] A. Iserles. *A first course in the numerical analysis of differential equations*. Cambridge University Press, 1996.
- [10] J. Jia and J. Huang. Krylov deferred correction accelerated method of lines transpose. *Journal of Computational Physics*, 227(3):1739–1753, 2008.
- [11] W. Y. Kong and V. Rokhlin. A new class of highly accurate differentiation schemes based on the prolate spheroidal wave functions. *Appl. Comput. Harmon. Anal.*, 2012. doi:10.1016/j.acha.2011.11.005.
- [12] H. J. Landau and H. O. Pollak. Prolate spheroidal wave functions, Fourier analysis and uncertainty II. *Bell System Tech. J.*, 40:65–84, 1961.
- [13] H. J. Landau and H. O. Pollak. Prolate spheroidal wave functions, Fourier analysis and uncertainty III. *Bell System Tech. J.*, 41:1295–1336, 1962.
- [14] Anita T. Layton and Michael L. Minion. Implications of the choice of quadrature nodes for Picard integral deferred corrections methods for ordinary differential equations. *BIT*, 45(2):341–373, 2005.
- [15] F.G. Lemoine, S.C. Kenyon, J.K. Factor, R.G. Trimmer, N.K. Pavlis, D.S. Chinn, C.M. Cox, S.M. Klosko, S.B. Luthcke, M.H. Torrence, et al. The development of the joint NASA GSFC and the National Imagery and Mapping Agency (NIMA) geopotential model EGM96. *NASA*, (19980218814), 1998.
- [16] K. Sandberg and K.J. Wojciechowski. The EPS method: A new method for constructing pseudospectral derivative operators. *J. Comp. Phys.*, 230(15):5836–5863, 2011.

- [17] J.M. Sanz-Serna. Runge-Kutta schemes for Hamiltonian systems. *BIT*, v. 28:877–883, 1988.
- [18] D. Slepian. Prolate spheroidal wave functions, Fourier analysis and uncertainty IV. Extensions to many dimensions; generalized prolate spheroidal functions. *Bell System Tech. J.*, 43:3009–3057, 1964.
- [19] D. Slepian. Some asymptotic expansions for prolate spheroidal wave functions. *J. Math. and Phys.*, 44:99–140, 1965.
- [20] D. Slepian. Prolate spheroidal wave functions, Fourier analysis and uncertainty V. The discrete case. *Bell System Tech. J.*, 57:1371–1430, 1978.
- [21] D. Slepian. Some comments on Fourier analysis, uncertainty and modeling. *SIAM Review*, 25(3):379–393, 1983.
- [22] D. Slepian and H. O. Pollak. Prolate spheroidal wave functions, Fourier analysis and uncertainty I. *Bell System Tech. J.*, 40:43–63, 1961.
- [23] H. Xiao, V. Rokhlin, and N. Yarvin. Prolate spheroidal wavefunctions, quadrature and interpolation. *Inverse Problems*, 17(4):805–838, 2001.